

## 2.5: Application: Number Systems and Circuits for Addition

Recall our how we write numbers in base 10:

$$\begin{aligned} 5,049 &= 5 \cdot 1000 + 0 \cdot 100 + 4 \cdot 10 + 9 \cdot 1 \\ &= 5 \cdot 10^3 + 0 \cdot 10^2 + 4 \cdot 10^1 + 9 \cdot 10^0 \end{aligned}$$

### Definition.

Any integer  $b > 1$  can be used as a base for a numbering system. A numbering system of base  $b$  has the digits  $0, 1, \dots, b - 1$ .

A **base 2 notation** or **binary notation**, uses the digits 0, 1. In binary, every integer is represented as sum of products of the form

$$d \cdot 2^n$$

where  $n \in \mathbb{Z}$  and  $d \in \{0, 1\}$ .

**Example.** Below is the binary representation for the integers 1 to 9:

$$\begin{array}{rclcl} 1_{10} & = & & 1 \cdot 2^0 & = & 1_2 \\ 2_{10} & = & & 1 \cdot 2^1 + 0 \cdot 2^0 & = & 10_2 \\ 3_{10} & = & & 1 \cdot 2^1 + 1 \cdot 2^0 & = & 11_2 \\ 4_{10} & = & 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 & = & 100_2 \\ 5_{10} & = & 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 & = & 101_2 \\ 6_{10} & = & 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 & = & 110_2 \\ 7_{10} & = & 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 & = & 111_2 \\ 8_{10} & = & 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 & = & 1000_2 \\ 9_{10} & = & 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 & = & 1001_2 \end{array}$$

**Converting binary  $\rightarrow$  decimal:**

To convert from binary to decimal, multiply each digit by its corresponding power of 2 and sum the results.

**Example.** Represent the following in decimal notation (base-10):

 $110_2$  $1011_2$  $11110_2$  $101011_2$

**Converting decimal  $\rightarrow$  binary:**

To convert from decimal to binary, we repeatedly divide by 2, and record the remainders.

**Example.**

$$\begin{aligned} 27_{10} &= 16 + 8 + 2 + 1 \\ &= 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 \\ &= 11011_2 \end{aligned}$$

**Example.** Represent the following in binary notation:

$$243_{10}$$

$$587_{10}$$

$$990_{10}$$

$$531_{10}$$

**Binary arithmetic:**

In binary arithmetic,  $10_2$  behaves similarly to 10 in decimal arithmetic.

**Example.** Add  $1101_2$  and  $111_2$  using binary notation.

**Example.** Subtract  $1011_2$  from  $11000_2$  using binary notation.

**Definition.**

**The 8-bit two's complement** for an integer  $a$  between  $-128$  and  $127$  is the 8-bit binary representation for

$$\begin{cases} a, & \text{if } a \geq 0 \\ 2^8 - |a|, & \text{if } a < 0. \end{cases}$$

Two's complement allows maximum representation for  $2^8$  integers with 8 binary digits.

**Example.** Below are a few integers represented in binary using 8-bit two's complement:

$$\begin{array}{ll} -128 \rightarrow 2^8 - |-128| = 128_{10} = 10000000_2 & 0 \rightarrow 0_{10} = 00000000_2 \\ -127 \rightarrow 2^8 - |-127| = 129_{10} = 10000001_2 & 1 \rightarrow 1_{10} = 00000001_2 \\ \vdots & 2 \rightarrow 2_{10} = 00000010_2 \\ -2 \rightarrow 2^8 - |-2| = 254_{10} = 10000000_2 & \vdots \\ -1 \rightarrow 2^8 - |-1| = 255_{10} = 11111111_2 & 127 \rightarrow 127_{10} = 01111111_2 \end{array}$$

**Example.** Find the 8-bit two's complement for the following:

$-46$

$42$

$120$

$-82$

**Two's complement of a negative integer:**

To find the decimal representation of the negative integer with a given 8-bit two's complement:

- Flip the bits
- Add 1
- Convert to base-10 and swap the sign

**Example.** Find the decimal representation of the integers with the following 8-bit two's complement:

 $11100101_2$  $11000000_2$ **Addition and Subtraction with Integers in Two's Complement Form:**

When performing binary addition on integers written in Two's Complement form, we discard any “carry” bit.

**Example.** Perform binary addition using the Two's Complement form of the following:

 $-87$  and  $-46$  $83$  and  $-55$

**Definition.**

**Hexadecimal notation** uses a **base 16 notation**. In hexadecimal, every integer is represented as sum of products of the form

$$d \cdot 16^n$$

where  $n \in \mathbb{Z}$  and  $d \in \{0, 1, \dots, 9, A, B, C, D, E, F\}$ .

Decimal	Hexadecimal	4-Bit Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

**Example.** Convert  $3CF_{16}$  to decimal notation.

**Example.** Convert  $B09F_{16}$  to binary notation.

**Example.** Convert  $100110110101001_2$  to hexadecimal notation.